Cost Reduction Through Application of Fieldbus Technology to Space Mission Operation and Control

Michael K. Jones
Senior Engineering Staff
NASA/Jet Propulsion Laboratory
Mission and Systems Architecture Section
Pasadena, California 91109

Elin L. Klaseen
Senior Research Engineer
SRI International
Information, Telecommunications, and Automation Division
Menlo Park, California 94025

Lee Neitzel
Senior Technologist
Fisher-Rosemount Systems, Inc.
Austin, Texas 78754

Keywords

FOUNDATION, fieldbus, interoperability, space mission, architecture, conformance

Abstract

For well over a decade, the international space research community has been laying the foundation for increased standardization of the interactions between spacecraft and their ground support systems. This paper reports on recent work by the Space Project Mission Operations Control Architecture (SuperMOCA) program, sponsored by the National Aeronautics and Space Administration (NASA) Space Operations Management Office (SOMO), in the definition of a formal Space Messaging Service (SMS) application protocol and of space mission device and system models based on the FOUNDATION ™* fieldbus architectural concepts, protocols, and products. While the FOUNDATION fieldbus effort grew out of needs within the process control industry, many of the architectural and performance requirements that had to be addressed by that community are similar to those that need to be addressed for remote space mission operations. These requirements include device interoperability, realtime scheduled and unscheduled bus access, physical resource constraints, and increasing device autonomy. In the era of dual-use development, the FOUNDATION fieldbus provides a solid architectural foundation from which to focus on the unique needs of space mission operations and control.

1. INTRODUCTION

The international space research community, under the guidance of the Consultative Committee for Space Data Systems (CCSDS), has been laying the foundation for increased standardization of the interactions between spacecraft and their ground support systems. The present CCSDS recommendations for packet telemetry, telecommand, and advanced orbiting systems are in widespread use throughout the world space community. More recently, the U.S. Space Communications Protocol Standards (SCPS) project sponsored by the U.S. National Aeronautics and Space Administration (NASA), the U.S. Department of Defense (DoD), and the United Kingdom Defence Research Agency (DRA), has been expanding “up the stack” to provide reliable transport and internetworking capabilities. With this basic ground-to-space transport profile in place, the Space Project Mission Operations and Control Architecture (SuperMOCA) program has been initiated with the goal of defining open international standards for space mission monitoring and control application interoperability. This paper presents background on the needs and requirements that have driven the SuperMOCA program standardization effort, the architectural approach applied by the program, and the outstanding issues that must be addressed in the coming years to meet the program goals.

---

*All product or company names mentioned in this document are the trademarks of their respective holders.

2. NEEDS AND REQUIREMENTS

Space projects must increasingly find new ways to reduce the cost of building, maintaining, and operating missions. Current methods are too expensive and result in unique, noninteroperable systems that require significant design reengineering. While new equipment and novel approaches may provide some relief, the SuperMOCA program has focused on the more fundamental need to reduce the level of effort and time (and thus the cost) required for the design, development, integration, operation, and maintenance phases of these complex systems.

Each of these system lifecycle phases requires some form of monitoring and control interaction (i.e., instrumentation) of the spacecraft engineering and science subsystems and their devices. Whether in the testbed environment or in a deployed system, all subsystems must provide a means for reporting health status, performing some amount of calibration and diagnostics, and executing commanded operations (e.g., guidance, navigation, or collection). For subsystems and intelligent devices with computing and storage capabilities, additional interaction will be required for the loading of software and tables of parameters, and the sequencing or invocation of software commands and procedures. In a deployed system, one or more onboard centralized controllers (i.e., executive or flight software, or command and data subsystems) will interact with the onboard devices, collecting and packaging data for the downlink to the ground systems, as well as receiving data and commands on the uplink for control or distribution to the spacecraft resources. This level of interaction with the ground can range from 24-hour, continuous interactions to complete autonomy, in which the spacecraft communicates with the ground system only in emergency situations.

During the development phase, a standalone host and input/output (I/O) acquisition interface cards are a first step in instrumenting and testing the physical components of the device through hardware and software transducer interfaces. However, as devices are integrated into the larger system, dedicated wiring to the signal interfaces of all devices becomes extremely complex to build and maintain. Backplane or bus network communication architectures are a means of reducing these costs. Because of the time-critical nature of some flight system operations, the selected communication architecture must provide reliable, guaranteed bus access in a millisecond-to-submillisecond time frame. To shorten the subsystem and system integration phase within these distributed, networked environments, devices should conform to standardized communication mechanisms for networked data access, and, from an external viewpoint, exhibit standardized behavior. Because of the lack of current standards for application data access (and the corresponding availability of cheap commercial products), space mission command and control must be reinvented to a significant degree for each mission, driving up the costs.

A requirement for a standardized communication interface to the physical resources of a device does not in turn require that the physical resources themselves be standardized. Device designers, particularly designers of scientific instruments, must have the freedom to design devices unique to their mission requirements, and must be constrained only to remain within the power, thermal, and mass budgets of the mission. As an example, the recently launched Cassini Orbiter includes an imaging device that contains both a wide-angle and a narrow-angle camera. Each camera has two filter wheels with selectable (by stepper motor) band-pass or band-reject filters. (These multiple filters are required for the study of the varying atmospheric conditions of Saturn.) Multiple exposure durations are controlled by pulsed output to a shutter device, to allow long exposures in the search for planetary acoustic oscillations. Similar elements (e.g., lenses, filter wheels, shutters, and charge coupled device sensor elements) can be found in imaging subsystems of the Galileo Orbiter or Mars Observer; the actual physical components and required ranges of operation may be considerably different, but the monitor and control capabilities are similar (e.g., discrete control of a stepper motor, on/off control of a heater, an analog temperature reading).

The above cost reduction requirements have resulted in three key design drivers for the SuperMOCA program:

- To enable software to be reused (and thereby reduce costs), spacecraft devices, subsystems, and system control software must consist of building block components. These components may range in granularity from a single I/O control point, to multiple I/O point controllers providing complex reasoning algorithms. The components must be simple yet comprehensive enough to be interfaced and mapped by device designers to the physical and logical components of their devices. Designers should also be able to extend the basic component classes when such extensions are needed.

- To speed up integration time (and thereby reduce costs), the communication interface to the device must be standardized. Standardization should be provided at several levels: the physical device level (for network and system initialization); the logical device level (for logical communication initialization and operation); the block level (for block mode initialization and operation); and finally at the parameter level (for parameter read/write access). Furthermore, the communications architecture should provide a means of establishing virtual monitor and control paths from ground-based applications or human operators to processes embedded within the onboard system.
- To enable a system to accommodate devices of varying levels of capability, and to be extensible in the future, the placement of the software components within the system, and the system's ability to accept new software components, should be design decisions and should not be constrained by the architecture.

3. APPROACH

The monitoring and control of systems in space, although unique in some features, is similar to that of industrial process control applications. While many process control communication solutions exist, the SuperMOCA program is examining and applying the Fieldbus Foundation (FF) consortium's FOUNDATION fieldbus technology to the problem of space mission operations and control. This technology was selected because of its solid and comprehensive architectural foundation, and because of the Fieldbus Foundation members' commitment to providing open, evolutionary solutions.

While the scope of the SuperMOCA program encompasses monitor and control loops internal to the spacecraft as well as those that span the ground and spacecraft systems and those wholly on the ground (e.g., control of the ground terminal resources), our first step in applying the FOUNDATION fieldbus standards has been to focus on the interactions wholly within the devices of the onboard system.[*] Past experience indicates that the interfaces of the onboard system will drive the interfaces that are exposed externally. As a second step, we have begun to define the space messaging services (SMS) necessary to support interactions with ground systems within an extended internetted environment, as well as mechanisms for the exchange of system-descriptive information (XDI). It is anticipated that these later developments will result in technology that can be transitioned back to the process control industry. Figure 1 depicts the SuperMOCA notion of virtual monitor and control paths and the communication and application architectural components required to support such access.

---

[*] A parallel SuperMOCA effort has been focused on the interactions between ground systems and ground terminal devices (e.g., antennas). A paper on this work was recently presented at the Satellite Command, Control and Networking American Defense Preparedness Association / National Security Industrial Association (ADPA/NSIA) conference. (Carrion, C. 1997. "An Implementation of a Commercial Messaging System Standard for Space Mission Applications," presented at the ADPA/NSIA Conference, Reston, Virginia [September]).

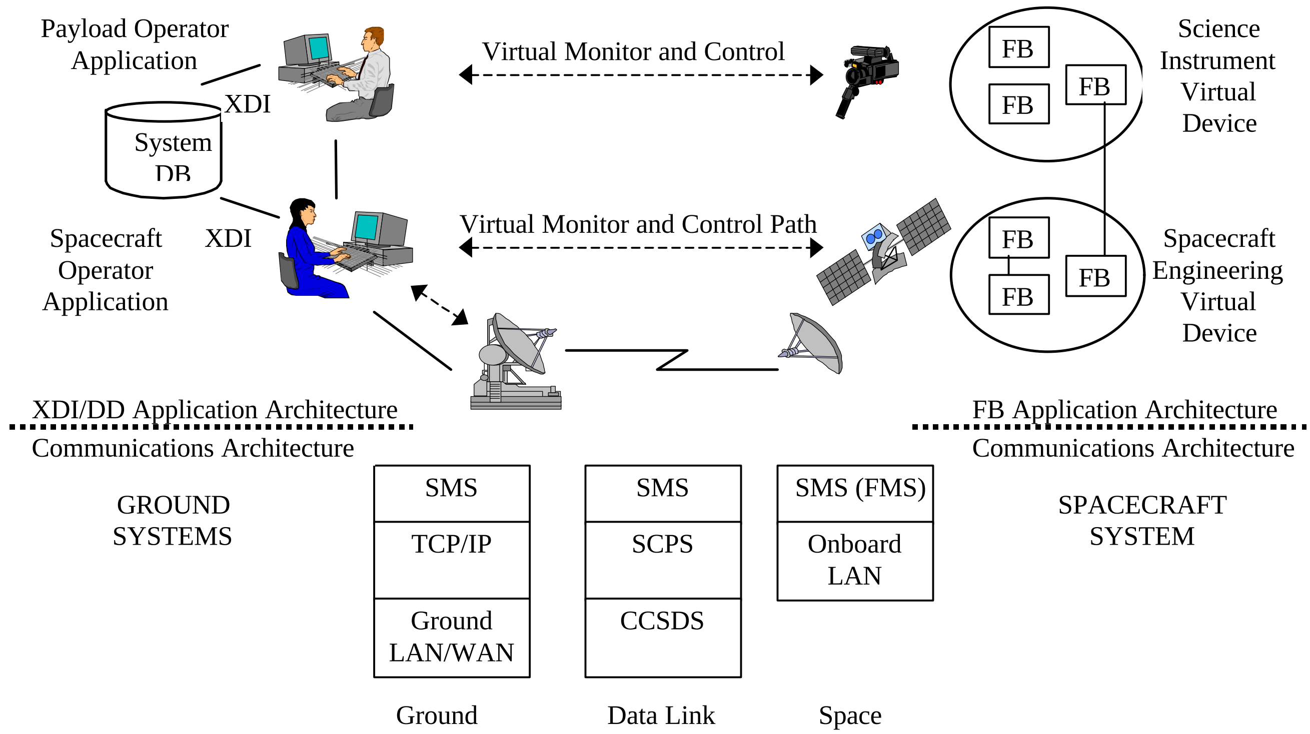

Figure 1. SuperMOCA Application and Communication Architecture

The following sections provide background on the FOUNDATION fieldbus architecture, and the SuperMOCA SMS and XDI progress.

3.1 FOUNDATION fieldbus

The FOUNDATION fieldbus architecture consists of a layered, standardized communications profile and accompanying standardized, function-block-based application software components. The architecture defines standardized mechanisms for additions, movements, and changes of devices within the network. Each of these elements is described below.

3.1.1 Function Block Applications

The heart of the FOUNDATION fieldbus standards is the specification for user applications. This specification is based on standard software building blocks. These blocks, termed function blocks, represent the basic automation functions performed by an application; these functions are as independent as possible of the specifics of I/O devices and the network. The function block architecture provides an essential, fundamental framework not only for application interoperability, but also for promoting the reuse of software within and across space missions.

> Each function block processes input parameters according to a specified algorithm and an internal set of contained parameters. They produce output parameters that are available for use within the same function block application or by other function block applications.[*]

Control strategies are defined through the linkage of the output and input parameters of function blocks within a device or across multiple devices. The initial function block classes include input, output, control, and calculate-type functions.

---

[*]FOUNDATION fieldbus. 1997. *FOUNDATION fieldbus Specification, Function Block Application Process*, Part 1, FF-890, Fieldbus Foundation (28 February).

Two additional block types describing the device and user applications are resource blocks and transducer blocks. Resource blocks describe characteristics of a device such as the device name, manufacturer, and serial number. Transducer blocks are defined to insulate function blocks from the specifics of the I/O devices such as sensors, actuators, and switches. Transducer blocks also perform functions such as calibration and linearization on I/O data to convert it to a device-independent (digital) representation. Through the licensing and use of existing function block software, device vendors can reduce their software development costs to those of producing only device-specific transducer blocks.

A virtual device is described to the external world through the collection and enumeration of parameters within these three block types through the use of a formal text-based language, termed the device description language (DDL). The vendor’s device description is compiled and stored in binary form by host and/or device applications. These applications can then be quickly built by utilizing the DDs as drivers for the devices.

3.1.2 Communication Stack

The Foundation fieldbus communications profile is a three-layer stack. The physical layer standard and current implementations are intended for low-power, low-bandwidth, intrinsic-safety environments. Additional higher bandwidth physical layers are currently being addressed by FF technical working groups. The data link layer (DLL) manages the access to the fieldbus through a deterministic centralized bus scheduler termed the link active scheduler (LAS). The LAS is a hybrid, token-based scheduler providing scheduled (synchronous) and unscheduled (asynchronous) bus access. The DLL provides peer or multipeer connection or connectionless oriented services, buffered and queued data transfer, and time distribution and synchronization. Multiple-link topologies are supported through a form of spanning tree bridging. Redundancy of the physical links, LAS, time publisher, and bridge ports is provided within the standard.

The third layer consists of the Fieldbus Access Sublayer (FAS) and the Fieldbus Message Specification (FMS). The FAS uses the scheduled and unscheduled services provided by the DLL to provide services, termed virtual communication relationships (VCRs), to the FMS. Three types of VCRs, have been defined to meet the needs of function block application interactions. These are

- Client/Server VCR—for queued, unscheduled, user initiated, one-to-one interactions between devices on a fieldbus
- Report Distribution VCR—for queued, unscheduled, user-initiated, one-to-many interactions
- Publisher/Subscriber VCR—for buffered, one-to-many, scheduled interactions.

The FMS describes the communication services, message formats, and protocol behavior needed to build messages for applications. The FMS provides the following services:

- Context management services—to establish/release VCRs and determine status of a virtual device
- Object dictionary services—for access to object description within the virtual device
- Variable access services—for read/write access to single parameters or lists of parameters
- Event services—for the reporting and management of alarms
- Upload and download services—for the upload/download of data and programs
- Program invocation services—for the execution of a program within a device.

The object dictionary provides an efficient, tagged means of describing the message data contents. All network-visible objects within a device have an associated object dictionary type description that indicates whether the object data type is elemental (e.g., float, unsigned 8-bit), an array of elemental types, or a structure.

3.1.3 System Configuration and Startup

System configuration is accomplished at four levels. Level 1 configuration consists of device-manufacturer-specific values for universal parameters such as device, model, and revision information. The device manufacturer supplies these values and a description of the block and parameter types used within the device in a text-based DD. Level 2 configuration is the level in which the network topology is defined. Device tags and data link addresses are determined, and the LAS and primary and secondary time masters are assigned. Level 3 configuration defines the distributed control application. At this stage, the I/O, alarm, and trending parameters of function blocks are linked through the assignment of VCR types. The function block and LAS schedules and macrocycles are defined. The spanning tree topology also is defined, including the definition of the forwarding and republishing tables. Level 4 configuration is the assignment of values to the operational parameters of the device.

Configuration at all levels can largely be accomplished off-line. On-line additions, modifications, and deletions can also be accomplished with minimal to no impact on the rest of the operating system. This standardization of the initialization of system parameters and entities such as the link and time masters represents a significant time savings during development and integration phases.

3.1.4 Interoperability and Conformance Testing

Any multivendor interoperability solution must include a means of testing product conformance to the standard. The FF has established interoperability testing centers and formal conformance test suites for the FOUNDATION fieldbus specifications. To date, one year after the introduction of the standards, five vendor communication stacks have passed the conformance tests. The FF's interoperability testing guarantees that products can provide a "plug and play" capability in a FOUNDATION-fieldbus-compliant system. As devices become commodities, interoperability enables the user to choose the best device for a specific control or measurement task, regardless of the manufacturer.

3.2 Space Messaging Service

The FOUNDATION specifications provide a fundamental framework for building space or ground systems composed of interacting devices. This framework must, however, be extended if it is to provide space-to-ground interoperability. The first extension is the definition of a 32-bit version of the FMS for operation over Transmission Control Protocol/Internet Protocol (TCP/IP)-based communication stacks. A prototype gateway device is being built to translate FMS requests from internetted remote users to the reduced stack fieldbus environment. While our initial testing will utilize standard UNIX and Windows/NT TCP/IP implementations, subsequent testing will utilize the SCPS stack developed by NASA, DoD, and DRA specifically for long-delay, limited-bandwidth, and potentially noisy-link space communications. This collection of protocols and gateways is called the SMS.

To reduce their mission operational costs, future missions must become increasingly autonomous, able to function for extended periods of time without connectivity to the ground. Space systems will thus require larger mass storage capacities, larger processing capacities for autonomous reasoning and actions, efficient upload and download mechanisms, and dynamic mechanisms for establishing connectivity to the ground on an as-needed, emergency basis. Depending on the level of memory management provided, the FMS domain upload and download services, or SCPS file transfer services, can be used for large data movement. Dynamic link establishment options are being explored within other NASA programs.

3.3 Exchange of Descriptive Information

A second major thrust and contribution of the SuperMOCA effort will be a general *Exchange of Descriptive Information* model for the identification and transfer of configuration information used to describe systems. In practical terms, descriptive information exchange is necessary when two or more databases exist that describe the same thing, and there is a need to transfer information from one database to another. This problem is immediately encountered when an attempt is made to share information across configuration databases. Current solutions are to use a single vendor product throughout the system, or to translate and convert information by hand. These solutions are either very costly or limit the customer's ability to migrate and utilize products from more than one vendor.

The XDI framework provides a structure for navigating through unique application database schemas by defining a common way of identifying and describing all system information. The challenge of such an approach is to define an information model that is sufficiently flexible to describe the structure of any given system (i.e., a classification structure) and extensive enough to characterize each component of that system (i.e., its data representation). With such a model in place, and with a supporting protocol, existing products can be modified to, at a minimum, provide access to their particular views of the system information. Future product architectures can capitalize on a single logical system description, but will be able to be designed and implemented within partitioned databases where necessary.

4. OPEN ISSUES

Several open issues remain to be resolved before a FOUNDATION-fieldbus-based SuperMOCA architecture can be deployed. Key among these is the need for space qualified communication components. Most current systems are built to use the MIL-STD-1553/1773 bus. While MIL-STD-1553 (1553) provides basic master/slave read/write services, it does not have a standardized application-layer protocol or application building blocks. Each mission typically re-invents these interface definitions. The primary advantage of 1553 is that it is already space qualified. For migration purposes, SuperMOCA is considering the definition of a 1553 FOUNDATION fieldbus access sublayer to support the FMS; however, future directions will ultimately require higher-speed busses.

Current FOUNDATION fieldbus products support only low-bandwidth applications. Higher bandwidth physical layer implementations will be required to accommodate some of the larger telemetry data sets from the science instruments. The FF has recently opened a technical working group to investigate the mapping of the DLL to Ethernet-based physical media.

Space science instruments are very complex, containing multiple distinct resources or components.[*] Implementation architectures could use the FOUNDATION fieldbus products to connect instrument component smart devices to controllers, or as a single controller with direct connections to simple component devices. Depending on the choice of implementation, the processing and memory capacities of current FOUNDATION fieldbus round cards and interface cards may not be sufficient. It is also possible that new multipoint I/O blocks may be defined to better model the device and to achieve better efficiency.

Space missions and space mission devices will increasingly rely on intelligent, autonomous operation. Complex decision support logic applications provide for the continuous execution of mission objectives and the routine operation of spacecraft. These applications must be reworked to utilize the FOUNDATION fieldbus function block paradigm. In particular, uniform mechanisms for data status handling and alarm management must be devised. The new wealth of information that these applications will make available will allow real-time data acquisition from devices for detecting anomalous behavior, parameter tuning functions for configuring and calibrating devices, and long-range trending functions to support the prediction of device failures.

---

[*] Klaseen, E. 1997. *Space Imaging Camera Component Model*, Technical Report ITAD-1485-TR-97-005, SRI International, Menlo Park, California (January).

5. CONCLUSION

The goal of the SuperMOCA program is to develop an integrated set of fully “open” (i.e., vendor-neutral) international standards for the processes associated with space mission command and control. Recognizing the complexity of the problem, the program has sought to utilize existing, open standards where appropriate. The FOUNDATION fieldbus specifications offer a significant body of work with accompanying commercial products to contribute to meeting those goals. By adopting the SuperMOCA approach, space mission architects can significantly reduce the cost of developing, integrating, operating, and maintaining missions. By cooperating with the commercial industry through an open consortium such as the Fieldbus Foundation, SuperMOCA can serve to orient both consumers (the international space community) and suppliers (private industry) toward future interoperable space mission command and control systems.

To assist in educating the broader NASA and DoD community, the SuperMOCA program has developed a demonstration system consisting of a commercial imaging camera, a Global Positioning System receiver, a radio receiver, and a custom search controller. This portable roadshow system was demonstrated as a part of the recent Instrumentation Society of America (ISA) ’97 Technical Exposition Innovators Theater. Additional information on the SuperMOCA program and progress can be obtained by contacting the authors, or from the SuperMOCA World Wide Web site (`http://supermoca.jpl.nasa.gov/supermoca`).

6. ACKNOWLEDGEMENTS

The authors would like to thank Adrian Hooke, Space Mission Operations Standardization Program Manager, NASA Space Operations Management Office for his unflagging support of the promotion of standards within the international space community; Dave Glanzer, Director of Technology Development for the Fieldbus Foundation, for his comments and support; and Sheldon Lloyd and Tom Boyd of Fieldbus, Inc., for their support of the development of a FOUNDATION fieldbus-based demonstration system for the SuperMOCA project.